# OemToChar

Carefully manage unit sizes and buffer bounds checking

Sean Barnum, Cigital, Inc. [vita[1]]



2007-04-02

## Part "Original Cigital Coding Rule in XML"

Mime-type: text/xml, size: 6569 bytes

| | |
|---|---|
| **Attack Category** | • Malicious Input |
| **Vulnerability Category** | • Buffer Overflow<br>• Multibyte Character |
| **Software Context** | • String Parsing |
| **Location** | |
| **Description** | The OemToChar family of functions does not do bounds checking and is subject to wide-character vulnerabilities.<br><br>The OemToChar routines translate OEM characters into standard ANSI characters. That is, for OEM char #252 (a superscript 'n' as in 2^n), the routine translates it to a normal 'n'. Note that this is not necessarily reversible.<br><br>The usage behavior is very nontypical and the routines do NOT behave the same as each other.<br><br>Replace OemToChar with OemToCharBuff. Use the 'length' field as the *number of characters to translate* (i.e., the length of the source string). The destination string must have at least enough space to store that string length plus the null character. Note: Unlike OemToChar, OemToCharBuff does NOT stop when it encounters the end of a null-terminated string. Furthermore, unless the length was provided properly, the routine will not copy the null byte and might not null terminate the string.<br><br>With narrow (single-byte) characters, the routine allows you to specify the source and destination string addresses the same, allowing for in-place translation.<br><br>The remaining "BUFF" functions have standard bounds checking issues if you pass in the wrong size:<br><br>OemToCharBuff<br>OemToCharBuffA<br>OemToCharBuffW<br>OemToAnsiBuff |

1. http://buildsecurityin.us-cert.gov/bsi-rules/35-BSI.html (Barnum, Sean)

| | |
|---|---|
| | OemToAnsiBuffA<br>OemToAnsiBuffW |
| **APIs** | |

| Function Name | Comments |
|---|---|
| OemToChar | |
| OemToCharA | |
| OemToCharBuff | |
| OemToCharBuffA | |
| OemToCharBuffW | |
| OemToCharW | |
| OemToAnsi | |
| OemToAnsiA | |
| OemToAnsiBuff | |
| OemToAnsiBuffA | |
| OemToAnsiBuffW | |
| OemToAnsiW | |

| | |
|---|---|
| **Method of Attack** | Be very wary using these functions. Their behavior is very nontypical.<br><br>The OemToChar function has a variety of problems:<br>* no bounds checking is done, so an attacker can trigger a buffer overflow<br>* differences between single-byte and wide-character behavior can be misused<br><br>Also, for OemToCharBuff:<br>* the wide-character versions accept a buffer length in _characters,_ not _bytes._ If the user passes in the wrong size, the routine may think the buffer is twice its actual size and therefore overrun the buffer. |
| **Exception Criteria** | |
| **Solutions** | |

| Solution Applicability | Solution Description | Solution Efficacy |
|---|---|---|
| Whenever there is a need to convert from OEM characters to ASCII. | Use OemToCharBuff instead of OemToChar. Ensure that all parameters are specified correctly. | Effective. |

| | |
|---|---|
| **Signature Details** | BOOL OemToChar(<br>LPCSTR lpszSrc,<br>LPTSTR lpszDst<br>); |

| | |
|---|---|
| **Examples of Incorrect Code** | `TCHAR dst[15]; // Buffer is too small`<br>`LPTSTR lpszDst = dst;`<br><br>`if (! OemToChar(TEXT("String containing OEM characters"), lpszDst)) { /* handle error */ }` |
| **Examples of Corrected Code** | `const TCHAR src[] = TEXT("String containing OEM characters");`<br>`LPCSTR lpszSrc = src;`<br>`TCHAR dst[30];`<br>`LPTSTR lpszDst = dst;`<br><br>`DWORD charsToConvert = strlen(lpszSrc) + 1; // should include terminating NULL`<br>`if (charsToConvert > sizeof(dst))`<br>`{`<br>`/* Handle error since buffer is not large enough */`<br>`} else {`<br>`if (! OemToCharBuff(lpszSrc, lpszDst, charsToConvert )) { /* handle error */ }`<br>`}` |
| **Source References** | • Rough Auditing Tool for Security (RATS)[2]<br>• http://msdn.microsoft.com/library/default.asp?url=/library/en-us/winui/winui/windowsuserinterface/resources/strings/stringreference/stringfunctions/oemtochar.asp[3] |
| **Recommended Resources** | • MSDN reference for OemToChar[4]<br>• MSDN reference for OemToCharBuff[5] |
| **Discriminant Set** | **Operating System**: • Windows<br>**Languages**: • C • C++ |

## Cigital, Inc. Copyright



1. mailto:copyright@cigital.com

The Build Security In (BSI) portal is sponsored by the U.S. Department of Homeland Security (DHS), National Cyber Security Division. The Software Engineering Institute (SEI) develops and operates BSI. DHS funding supports the publishing of all site content.